

Robotic navigation algorithm with machine vision

César G. Pachón-Suescún, Carlos J. Enciso-Aragón, Robinson Jiménez-Moreno

Faculty of Engineering, Nueva Granada Military University, Bogotá D.C., Colombia

Article Info

Article history:

Received Jun 7, 2019

Revised Oct 11, 2019

Accepted Oct 20, 2019

Keywords:

Embedded System

Morphological filters

Robotic navigation

ABSTRACT

In the field of robotics, it is essential to know the work area in which the agent is going to develop, for that reason, different methods of mapping and spatial location have been developed for different applications. In this article, a machine vision algorithm is proposed, which is responsible for identifying objects of interest within a work area and determining the polar coordinates to which they are related to the observer, applicable either with a fixed camera or in a mobile agent such as the one presented in this document. The developed algorithm was evaluated in two situations, determining the position of six objects in total around the mobile agent. These results were compared with the real position of each of the objects, reaching a high level of accuracy with an average error of 1.3271% in the distance and 2.8998% in the angle.

*Copyright © 2020 Institute of Advanced Engineering and Science.
All rights reserved.*

Corresponding Author:

Robinson Jiménez Moreno,
Mechatronics Engineering Program, Faculty of Engineering,
Nueva Granada Military University,
Carrera 11 #101-80, Bogotá D.C., Colombia.
Email: robinson.jimenez@unimilitar.edu.co

1. INTRODUCTION

Currently, in the field of robotics, it can be found a series of methods to map a specific work area and, from this data, perform a processing according to the application in which the robotic agent will be developed. It is essential to extract this data accurately because the movement of the robotic agent depends on this data as seen in [1], where three robots map a labyrinth together in order to solve it. Most of the algorithms developed today are based on a system of integrated sensors, such as for example ultrasonic [2] or laser as is the case of [3]. At the national level, some examples of these situations can be seen [4-5], which generates a series of limitations when dealing with certain situations, such as distinguishing between two different types of objects. Since they are low-cost systems and that have extensive documentation in terms of instrumentation and mathematical modeling as mentioned in [6], they become the primary choice. On the other hand, there are algorithms based on a global camera as implemented in [7], which can cause that in environments where it is not possible to use a camera in that position, strategies that may increase the complexity of the system or restrict its functionality should be sought. Focusing studies on the design of environmental mapping algorithms and identification of this allows that algorithms such as the one presented in [8], where a trajectory planning algorithm is designed in a virtual environment, can be implemented in real environments.

This article proposes an alternative method to solve this problem that will be focused on the implementation of individual mobile agents whose task will be to identify specific objects within an established work area. To do this, an algorithm based on machine vision techniques is designed and through experimental tests, the necessary relationships are determined to make possible the equivalence between the real locations of each object versus that calculated by the algorithm.

In the state of art, many works about mobile robots are done. The main idea is making its auto self-driving [9, 10] using planning trajectories for this task, in 2D [11] and 3D [12] environments, considering energy-aware [13], terrains characteristics [14] and implementing optimization methods [15]. However,

machine vision systems are very useful for auto self-driving, to control the mobile robot [16] and avoid obstacles [17], such as presented in the present work. The article is divided into four main parts, the first part presents some theoretical foundations necessary for the understanding of the other stages. The second one focuses on the materials and methods, where the elements used for the tests and the calculations made for the detection of the objects of interest are shown. The third part shows the results obtained and two examples of cases in which the algorithm was tested. Finally, the conclusions regarding the designed algorithm are presented.

2. THEORETICAL FRAMEWORK

The algorithm is mostly developed on the OpenCV libraries for Python, since being a mobile agent independent of an external console, these software tools are a main alternative to be applicable in an embedded system. For the development of the algorithm, the fundamental bases of image processing were taken into account.

2.1. Color filters

Color filters are those that allow a specific color to be identified within a digital image, generally, these filters have a lower and upper range by which they limit which color or colors are those that are to be determined within the image. These ranges are defined according to a particular color scale, there are a wide variety of color scales, among which the most common are RGB and HSV, but each scale has its own characteristics that make each one have different applicability [18]. Table 1 shows the main advantages and disadvantages of the three color models that were considered to develop the algorithm.

Table 1. Advantages and disadvantages of three color models, based on [18]

Model	Advantages	Disadvantages
RGB	-It is used in the video screen due to its additive properties. -It is considered as a computationally practical model.	-It is not useful for specifying objects and recognizing colors. -It is difficult to determine a specific color.
HSV	Colors easily defined by human perception, unlike RGB.	Indefinite achromatic tone points are sensitive to deviations of RGB values and hue instability, due to the angular nature of the characteristic.
HSL	-The chrominance components (H and S) are associated with the way humans perceive, it is perfect for image processing applications. -The component (Hue) can be used to perform the segmentation process instead of the three components that make up the model.	-The indefinite achromatic tone points are sensitive to deviations of RGB values and tonality instability, due to the angular nature of the characteristic. -It is not uniform.

2.2. Morphological filters

These kinds of filters are commonly used in machine vision algorithms, they can perform different tasks depending on the filter applied, either eliminate the noise in an image [19] or identify the geometric structure of a given object [20]. It should be noted that this kind of filters is applied only on binarized images, i.e. images in which only the absolute white or black color is given, that is equivalent to 1 and 0, respectively. The morphological filters are theoretically an n-dimensional matrix whose structuring element can be circular or square, or even irregular, which can vary depending on the treatment that it is wanted to perform or the characteristics wanted to extract from the image, such as those observed in [21, 22].

In Figure 1, an example of how the most common morphological filters that are frequently used in applications related to object recognition work can be seen. On one hand, erosion is a matrix operation between pixels whose function is to reduce the number of white pixels by evaluating the proximity of each of them to the black pixels, depending on the structuring element see Figure 1b. On the other hand, the dilation operation works in a completely opposite way to the erosion, reason why the quantity of white pixels increases see Figure 1c.

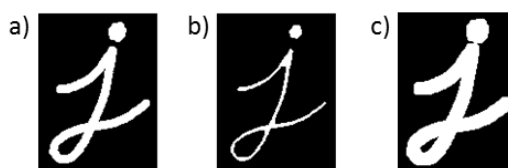


Figure 1. Morphological filters. (a) Original image, (b) erosion filter applied to the original image and (c) dilation filter applied to the original image [23]

3. MATERIALS AND METHODS

The algorithm developed is framed in a mobile robotics project, for this reason, there are a series of conditions for the development of the same, including the camera is in the structure of the agent in the lower frontal part, the agent has an embedded Raspberry Pi 3 system [24], which is responsible for performing all the necessary calculations for the different algorithms. The work area is 2 m² on flat terrain, but may have slight changes in lighting and the objects to be identified will be uniform cubic structures of magenta color. It should be noted that the algorithm was developed taking into account these guidelines, but this can be implemented in other types of robotic agents and work areas of different dimensions.

The agent must identify all the objects of interest within the work area, the identification algorithm starts taking 6 captures at 60 degree intervals, since the focus of the camera (Raspberry Pi Camera V2) has an approximate focus of 66 degrees, thus covering the entire perimeter around the mobile agent. As a first step, it is necessary for the agent to start taking captures of their environment, in this case, they were made with a resolution of 640x480 pixels. In Figure 2, one of these images can be observed, with which the procedure to identify the object will be explained later.



Figure 2. Original capture of the work area

For the identification of the objects of interest present in the image, it is necessary to propose a series of filters that allow determining with precision and accuracy in which position they are with respect to the mobile agent. In the majority of the applications related to machine vision and robotics, it is pertinent to comply with these two parameters to a lesser or greater extent, depending on the application to which it is focused, for this particular case, the values obtained must have a minimum error, given that these values are what will determine how the mobile agent should move within the work area without crashing and reaching a certain point. Based on this, a color filter is implemented that allows the use of the algorithm even when there are slight changes in ambient lighting. The color scale that best suited the needs was the HSL see Figure 3.

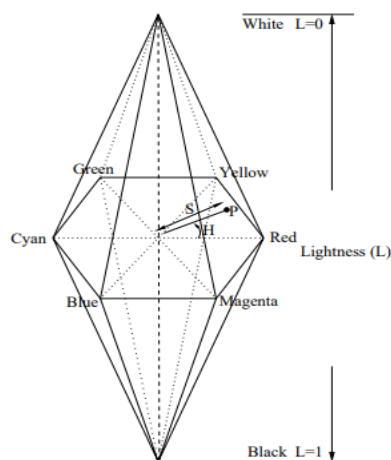


Figure 3. HSL color scale [25]

From this chart, the following ranges were defined for each color parameter:

$$\begin{aligned} H_1 &= 260 \text{ to } H_2 = 310 \\ S_1 &= 0 \text{ to } S_2 = 0.75 \\ L_1 &= 0.47 \text{ to } L_2 = 1 \end{aligned}$$

These parameters in OpenCV use other ranges that are from 0 to 180 for H and from 0 to 255 for both S and L, so the ranges that are finally applied in the programming are the following:

$$\begin{aligned} H_1 &= 130 \text{ to } H_2 = 155 \\ S_1 &= 0 \text{ to } S_2 = 190 \\ L_1 &= 120 \text{ to } L_2 = 255 \end{aligned}$$

Applying this color filter, it is obtained the result observed in Figure 4a. In the image obtained, small groups of pixels can be observed that are not part of the object, for that reason, they are considered as noise, being eliminated by morphological filters. First, an erosion operation is implemented with a 4x4 square structuring element see Figure 4b and finally, a dilatation operation with a 4x4 square structuring element to try to recover the original dimensions of the object. Once these morphological filters are applied, a remarkable change can be observed in Figure 4c.



Figure 4. Application of morphological filters. a) Binarized original image. b) Application of the erosion filter. c) Application of the dilation filter.

When all the identified segmented color figures are found, another problem must be faced: it is possible that there are objects of the same color near the work area, but with different shapes that would not correspond to the defined objects of interest, therefore, it is necessary to implement an additional filter that allows discriminating other objects that may have the same color. The filter that is decided to be implemented is based on the forms, which works in the following way: from Figure 4, the contours of the different elements present in the image are extracted. With the points that make up the contours, each of them is replaced forming line segments, finally, a number of lines and certain intersections are obtained. Therefore, it is useful to identify when it is an object of interest given that for this case they are cubes and in the image that is obtained through the treatment of images, an element with four edges is observed see Figure 5.

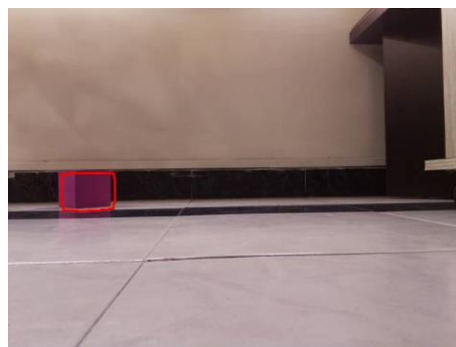


Figure 5. In the initial picture as a visual test, the contours of the objects identified as objects of interest are indicated.

When it is possible to determine that the objects present in the image are actually cubes, it is sought to apply the ratio of pixels to metric units to determine the distance to which the different objects present in the image are located. This relationship was established experimentally by taking capture of a cube aligned on the X-axis with the camera at various known distances. Additionally, the lower and upper-end points on the Y-axis were found for the object in the image, this in order to determine how many pixels represented the height of the cube, since this dimension remains invariant for the camera independently from what angle the image was taken. From these experimental data, a graph with its respective trend line and the equation that describes it were obtained see Figure 6.

From this data, a mathematical equation that relates the pixels with the distance from the camera to the object in a precise and exact way was obtained. Based on the image processing and taking into account the calculated ratio between metric units and pixels captured by the camera, the position of each object of interest is calculated using the mobile agent as a reference point see Figure 7. Based on Figure 7, a series of equations are proposed in order to calculate the polar coordinates of the object with respect to the agent in metric units, where $pix=height$ in pixels of the figure. Ca refers to the adjacent leg generated from the focus of the camera to the visible face of the object of interest. Therefore, equation 1 describes the value of the variable Ca .

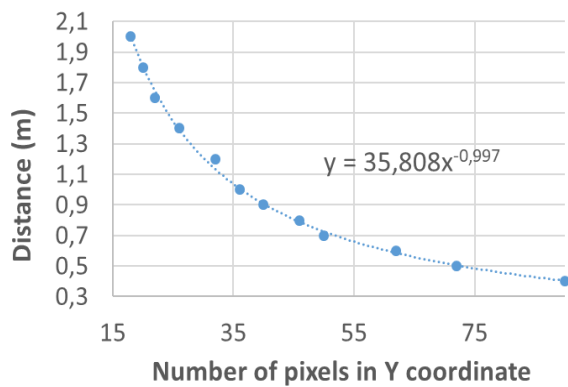


Figure 6. Graph of relationship between number of pixels and real distance to the cube in meters with exponential trend line

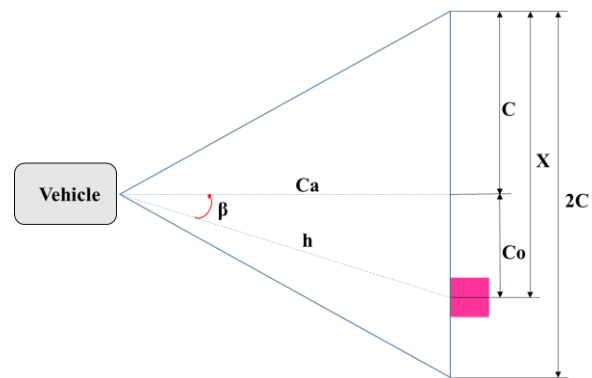


Figure 7. Illustration of the notation used for the calculations made

$$Ca = 35.808 * pix^{-0.997} \tag{1}$$

C is the opposite leg of the center of the camera see (2).

$$Ca = \tan(33) C \tag{2}$$

X_{coord} is the X coordinate in pixels to the center of the visible face. Taking into account that in the X-axis the image has 640 pixels, a conversion of pixels to metric units is made see (3)

$$X = \frac{(X_{coord})(2C)}{640} \tag{3}$$

Co is the distance from the center of the camera's focus to the center of the object see (4)

$$Co = X - C \tag{4}$$

Once the values of the adjacent leg and the opposite leg have been calculated, the polar coordinates are calculated with equations 5 and 6, taking into account that A refers to the angle that the agent has rotated between each capture.

$$\beta = \tan^{-1}\left(\frac{Co}{Ca}\right) + A \tag{5}$$

$$h = \sqrt{Co^2 + Ca^2} \quad (6)$$

Finally, the possible case is raised where one of the objects of interest can be captured in more than one image, so the mobile agent would interpret that there are more cubes than the real amount that is in the work area. To avoid this drawback, a filter is defined consisting of the condition that if two consecutive cubes with an angle difference of less than 10 degrees and a distance of ± 5 cm are detected in two consecutive captures, they are treated as the same cube, thus removing both from the list and determining an average between the two calculated positions. If, on the contrary, one of these two conditions is not met, they will be different cubes and both positions will be maintained in the list of objects. In the list of objects, it can be observed the polar coordinates of each one with respect to the mobile agent, taking into account that zero degrees starts aligned with the camera in the first capture and will increase in counterclockwise direction until completing the 360 degrees.

4. RESULTS AND DISCUSSION

What is sought is that the algorithm is generic, due to this, the number of objects in the work area can become n , where $n \rightarrow \infty$, for this reason, for the tests, a specific sample size is not established. The algorithm was put to tests in a real and controlled environment with an area of $2m^2$, where 3 cubes were randomly distributed within the work area, in the center the mobile agent was located and it was looked to check the measurements given by the algorithm to compare them with the real distances in which each cube was located. In Figure 8, the first case studied is observed.



Figure 8. First case study in a real work area.

Once the objects were located in random places, the algorithm was executed, and the results obtained were compared with the real measurements that were taken experimentally, calculating the approximate error as shown in Table 2.

Table 2. Comparison of the measurements taken from the algorithm with the real measurements for the first case study

Object	Angle		
	Algorithm	Real (approx.)	% Error
1	29.4435	30	1.8550
2	142.1999	147	3.2653
3	282.4787	281	0.5262
Distance (cm)			
Object	Distance (cm)		
	Algorithm	Real (approx.)	% Error
1	33.0166	31.5	4.8146
2	42.6121	40.8	4.4414
3	31.7967	32.4	1.8620

As shown in the previous table, the error is less than 5%. This represents that the level of precision and accuracy with which the algorithm detects the distance of objects from the agent is feasible for its implementation in the established work environment. Additionally, a second case was posed where the objects are positioned in such a way that the same cube is observed in two different frames, this in order to verify that the data taken by the algorithm contains the correct amount of information and that they effectively approximate the real values. In Figure 9, the new distribution of the objects within the work area can be observed. Again, the results were tabulated in order to obtain the percentage of error between the real measurement and that calculated by the machine vision algorithm, as can be seen in Table 3. Once the data of the two tests have been recorded, a maximum error in distance of 4.8146% can be evidenced, which is equivalent to 1.5166 cm, and 3.2653% error in angle, equivalent to 4.8 degrees. On the other hand, the average error in distance is 1.3271% and in angle, it is 2.8998%, which would allow implementing this algorithm without compromising the correct operation and mobility of the robot within the work area.



Figure 9. Second case study in a real work area

Table 3. Comparison of the measurements taken from the algorithm with the real measurements for the second case study

		Angle		% Error
Algorithm	Real (approx.)	Real (approx.)		
1	96.2896	96		0.3017
2	233.3020	230		1.4357
3	267.5392	266		0.5786
		Distance (cm)		% Error
Algorithm	Real (approx.)	Real (approx.)		
1	28.1190	27.5		2.2509
2	18.8660	19.2		1.7395
3	37.2338	36.4		2.2907

5. CONCLUSION

The developed algorithm is a valid starting point for tracking applications in the field of robotics that can be focused on tasks of grouping and evasion, since it allows identifying specific objects and, from these data, can determine how to maneuver or interact with them. It should be noted that the implementation of the algorithm has a relatively high cost compared to algorithms based on ultrasonic sensors, mainly due to the implementation of a camera and, in this case, an embedded system for its management. On the other hand, by implementing these two tools in a mobile agent, an average error of 1.3271% for the distance measurement and 2.8998% for the measurement of angles was obtained.

In comparison with algorithms developed with the help of a global camera, this has the advantage that it avoids the implementation of a communication system between the mobile agent and an external terminal that performs image processing. In addition, this type of architecture allows the applicability of this algorithm in tasks whose environments do not allow the use of a globalized camera. Although the algorithm allows to identify the polar coordinates of the objects of interest around it, it is necessary to design additional strategies to identify the elements when it is impossible to detect them in the first capture sampling of the work area, either by an obstacle, irregularities in the field or an external agent

ACKNOWLEDGEMENTS

The authors are grateful to the Nueva Granada Military University for the support given in the development of this work.

REFERENCES

- [1] Rodríguez, et al., 2014, "Mapeo de Laberintos y Búsqueda de Rutas Cortas Mediante Tres Mini Robots Cooperativos," *Politécnica*, vol. 34, no. 2, pp. 101-106, 2014.
- [2] M.O. Moussa, A. Moussa, and N. El-Sheimy, "Multiple ultrasonic aiding system for car navigation in GNSS denied environment," In *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pp. 133-140, 2018.
- [3] M. Pierzchała, P. Giguère, and R. Astrup, "Mapping forests using an unmanned ground vehicle with 3D LiDAR and graph-SLAM," *Computers and Electronics in Agriculture*, vol. 145, pp.217-225, 2018.
- [4] G. Acosta, et al., "Una Arquitectura de Agente Robótico Móvil para la Navegación y Mapeo de Entornos de Trabajo," *CISCI*, 2008.
- [5] O. Zapata, J.A. Jiménez and G.A. Acosta, "Diseño de un Esquema de Coordinación de Comportamientos para la Navegación de una Plataforma Robótica Móvil," [online]. Available at: <http://www.bdigital.unal.edu.co/12437/1/1020415847.2014.pdf>. Consult date: Jun 10, 2019.
- [6] G.A. Acosta-Amaya, J.A. Jiménez and D.A. Ovalle Carranza, "Ambiente Multi-Agente Robótico para la navegación colaborativa en escenarios estructurados," [online]. Available at: http://www.bdigital.unal.edu.co/2533/1/71677978.2010_1.pdf. Consult date: Jun 10, 2019.
- [7] K. Murakami, et al., "Cooperative soccer play by real small-size robot," In *Robot Soccer World Cup, Springer, Berlin, Heidelberg.*, pp. 410-421, 2003.
- [8] C.G. Pachón-Suescún, C.J.E. Aragón, M.A.J. Gómez, and R. Jimenez-Moreno, 2017, "Obstacle Evasion Algorithm for Clustering Tasks with Mobile Robot," In *Workshop on Engineering Applications. Springer, Cham*, vol. 742, pp. 84-95, 2017.
- [9] Ayman A Abu Baker, Yazeed Ghadi, "Mobile robot controller using novel hybrid system," *International Journal of Electrical and Computer Engineering IJECE*, ISSN: 2088-8708. Vol 10 No1. pp 1027-1034, 2019.
- [10] S. George Fernandez, K. Vijayakumar, R Palanisamy, K. Selvakumar, D. Karthikeyan, D. Selvabharathi, S. Vidyasagar, V. Kalyanasundhram, "Unmanned and autonomous ground vehicle," *International Journal of Electrical and Computer Engineering IJECE*, ISSN: 2088-8708. Vol 9 No5. pp 4466-4472, 2018.
- [11] R. J. Moreno and D. Jorge Lopez, "Trajectory planning for a robotic mobile using fuzzy c-means and machine vision," *Symposium of Signals, Images and Artificial Vision - 2013: STSIVA - 2013*, Bogota, 2013, pp. 1-4. doi: 10.1109/STSIVA.2013.6644912.
- [12] R. Jiménez Moreno and L. Brito M., "Planeación de trayectorias para un móvil robótico en un ambiente 3D," *2014 IEEE Biennial Congress of Argentina (ARGENCON)*, Bariloche, 2014, pp. 125-129. doi: 10.1109/ARGENCON.2014.6868483.
- [13] Shirin Rahmanpour, Reza Mahboobi Esfanjani, "Energy-aware planning of motion and communication strategies for networked mobile robots," *Information Sciences*, Volume 497, 2019, Pages 149-164, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2019.05.034>.
- [14] Weihua Li, Zhencai Li, Yiqun Liu, Liang Ding, Jianfeng Wang, Haibo Gao, Zongquan Deng, "Semi-autonomous bilateral teleoperation of six-wheeled mobile robot on soft terrains," *Mechanical Systems and Signal Processing*, Volume 133, 2019, 106234, ISSN 0888-3270, <https://doi.org/10.1016/j.ymsp.2019.07.015>.
- [15] Qi-bin Zhang, Peng Wang, Zong-hai Chen, "An improved particle filter for mobile robot localization based on particle swarm optimization," *Expert Systems with Applications*, Volume 135, Pages 181-193, ISSN 0957-4174, 2019, <https://doi.org/10.1016/j.eswa.2019.06.006>.
- [16] Anna Annusewicz, "The use of vision systems in the autonomous control of mobile robots equipped with a manipulator," *Transportation Research Procedia*, Volume 40, Pages 132-135, ISSN 2352-1465, 2019, <https://doi.org/10.1016/j.trpro.2019.07.022>.
- [17] F. Espinosa, M. R. Jiménez, L. R. Cárdenas and J. C. Aponte, "Dynamic obstacle avoidance of a mobile robot through the use of machine vision algorithms," *Symposium of Signals, Images and Artificial Vision - 2013: STSIVA - 2013*, Bogota, pp. 1-5, 2013, doi: 10.1109/STSIVA.2013.6644903.
- [18] N.A. Ibraheem, M.M. Hasan, R.Z. Khan, and P.K. Mishra, "Understanding color models: a review," *ARPN Journal of science and technology*, vol. 2, no. 3, pp.265-275, 2012.
- [19] R. Farnoosh, M. Rahimi, and P. Kumar, "Removing noise in a digital image using a new entropy method based on intuitionistic fuzzy sets," In *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1328-1332, 2016.
- [20] S. Azam, and M.M. Islam, "Automatic license plate detection in hazardous condition," *Journal of Visual Communication and Image Representation*, vol. 36, pp.172-186, 2016.
- [21] E. Dougherty, "Mathematical morphology in image processing," *CRC press*, 2018.
- [22] X. Zhang, W. Qi, Y., Cen, H., Lin, and N. Wang, "Denosing vegetation spectra by combining mathematical-morphology and wavelet-transform-based filters," *Journal of Applied Remote Sensing*, vol. 13, no. 1, p.016503, 2019.

- [23] OpenCV, "OpenCV Documentation," [online], Available at: http://docs.opencv.org/2.4/doc/tutorials/imgproc/erosion_dilatation/erosion_dilatation.html. Consult date: jun 10, 2019.
- [24] Foundation Raspberry Pi, "Raspberry Pi", [Online], Available at: <http://www.raspberrypi.org/>. Consult date: jun 10, 2019.
- [25] K.N. Plataniotis, and A.N. Venetsanopoulos, "Color image processing and applications," *Springer Science & Business Media*, 2013.

BIOGRAPHIES OF AUTHORS



César Giovany Pachón Suescún was born in Bogotá, Colombia, in 1996. He received his degree in Mechatronics Engineering from the Pilot University of Colombia in 2018. Currently, he is studying his Master's degree in Mechatronics Engineering and working as Research Assistant at the Nueva Granada Military University with an emphasis on Robotics and Machine Learning. E-mail: u3900259@unimilitar.edu.co



Carlos Javier Enciso Aragón was born in Bogotá, Colombia, in 1996. He received his degree in Mechatronics Engineering from the Pilot University of Colombia in 2018. Currently, he is studying his Master's degree in Mechatronics Engineering and working as Research Assistant at the Nueva Granada Military University with an emphasis on Robotics and Machine Learning. E-mail: u3900256@unimilitar.edu.co



Robinson Jiménez Moreno was born in Bogotá, Colombia, in 1978. He received the Engineer degree in Electronics at the Francisco José de Caldas District University - UD - in 2002. M.Sc. in Industrial Automation from the Universidad Nacional de Colombia - 2012 and Ph.D. in Engineering at the Francisco José de Caldas District University - 2018. He is currently working as a Professor in the Mechatronics Engineering Program at the Nueva Granada Military University - UMNG. He has experience in the areas of Instrumentation and Electronic Control, acting mainly in Robotics, control, pattern recognition, and image processing. E-mail: robinson.jimenez@unimilitar.edu.co